

REMARKS/ARGUMENTS

Claims 1-31 are pending in the present application. By the present response, claims 8-14 have been amended. Reconsideration of the claims is respectfully requested.

I. 35 U.S.C. § 103, Obviousness: Claims 1, 6, 7, 8, 13, 14, 15, 20-22, 27 and 28

Claims 1, 6, 7, 8, 13, 14, 15, 20-22, 27 and 28 stand rejected under 35 U.S.C. § 103 as being unpatentable over **Hawkins et al.**, Method and Apparatus for Communicating Information Over Low Bandwidth Communications Networks, U.S. Patent No. 6,343,318 (January 29, 2002) (hereinafter “**Hawkins**”) in view of Java Servlet Programming by Jason Hunter, published November 1, 1998 (hereinafter “**Hunter**”), and in further view of **Ginter**, Systems and Methods for Secure Transaction Management and Electronic Rights Protection, U.S. Patent No. 5,892,900 (April 6, 1999) (hereinafter “**Ginter**”). This rejection is respectfully traversed.

The rejection states:

Regarding independent claim 1 Hawkins discloses transforming a file into a pervasive computing device specific file. Hawkins recites: “The proxy server 180 responds to requests by wireless clients 405 ... The proxy server 180 carries most of the burden of bringing the information from the Internet 190, converting it ... and transferring it to the wireless client 405 ...

Hawkins discloses in Figure 1 receiving a request for the original file at a server ... the file being stored at the server. As shown in Figure 1 ... the server is shown at reference sign 140 (described as ‘Web Server’). The server is shown storing a document at reference sign 144 (described as “HTML Page”).

Hawkins discloses performing the conversion process steps at the server. Hawkins recites: “server 180 carries most of the burden of bringing the information from the internet 190, converting it to wireless client 405 compatible CTP and CML formats, and transferring it to the wireless client 405 over the wireless network’ (column 261, lines 1 8-23).

Claim 1 is directed toward the file being a Java Server Page (JSP) file, JSP files differ from plain markup language files in that JSP markup includes executable code for program execution, rather than just tags for formatting control. Hawkins discloses that the file contains executable program code Hawkins recites: “Alternatively, some programs are customized for accessing specific information from particular web sites. Examples of these programs are Java applets that reside on the client or are served to the client by a server” (column 3, lines 14-17),

However Hawkins fails to explicitly recite JSP. Hunter teaches that Java applets that are received from a server (called servlets) are the same as Java Server Pages. Hunter recites: Just as this book was going to press, Sun announced a new way to use servlets, called Java Server pages (commonly, but not officially referred to as JSP). JSP’s functionality and syntax bear a remarkable resemblance to Active Server Pages “ASP”) (first paragraph of section 2.6). Therefore, it would have been obvious, to one of ordinary skill in the art, at the time the invention was made, to use JSP style program code, as taught by Hunter, in the program code enabled files of Hawkins.

because Hunter teaches that the functionality and syntax of JSP's resemble other server page languages. ...

Hawkins and Hunter fail to disclose the masking and unmasking of specific tags in the conversion process Ginter teaches the use of masking tags. Ginter recites: "UDEs 1200 are preferably encrypted using a site specific key once they are loaded into a site. This site-specific key masks a validation tag" (column 150, lines 35-37).

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time the invention was made, to combine the masking of tags as taught by Ginter with the transformation of files for pervasive computing devices as taught by Hawkins and Hunter in order to "maintain the integrity, availability, and/or confidentiality of such information and processes related to such use" (Ginter, column 1, lines 13-15).

Office Action mailed June 7, 2006, pages 3-6, emphasis added

The determination of "nonobviousness" is made after establishing the scope and content of prior art, the differences between the prior art and the claims at issue, and the level of ordinary skill in the pertinent art. *Graham v. John Deere*, 383 U.S. 1 (1966). In addition, all limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 21 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). A *prima facie* case of obviousness is established when the teachings of the prior art itself suggest the claimed subject matter to a person of ordinary skill in the art. *In re Bell*, 991 F.2d 781, 783, 26 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1993). The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990).

Claim 1 recites:

1. A method of transforming an original Java Server Page (JSP) file into a pervasive computing (PvC) device specific JSP file, comprising:
 - receiving a request for the original JSP file at a server, wherein the request is sent from the PvC device, and wherein the original JSP file is stored on the server;
 - parsing the original JSP file for JSP tags;
 - masking the JSP tags;
 - converting non-masked tags in the original JSP file into PvC device specific format tags;
 - unmasking the JSP tags; and
 - storing a transformed JSP file containing the PvC device specific format tags and the JSP tags, wherein the parsing, masking, converting, unmasking and sending steps are performed by the server.

A proper *prima facie* case of obviousness has not been made against claim 1 because the combination of references does not disclose all of the elements of the claim, nor do the references suggest the desirability of their combination. These problems will be discussed separately.

Claimed Features

The rejection asserts that “*Hawkins discloses transforming a file into a pervasive computing device specific file*”, but does not show that **Hawkins** discloses the specific steps of the invention recited in claim 1. The elements of claim 1 that are not met by the combination of **Hawkins**, **Hunter**, and **Ginter** are numerous and include the following elements that will be specifically addressed: (1) the combination does not disclose a single server performing the recited steps, (2) the combination does not disclose that the file to be converted contains JSP pages or their equivalent, and (3) the combination does not disclose the masking and unmasking of JSP tags.

Single Server not Shown Performing the Recited Steps

Claim 1 recites that a single server provides both the content of the web page (*the original JSP page is stored on the server*) and converts the page into a format compatible with the PvC device (*converting ... tags*). An advantage to this approach is that a content provider can provide access for portable devices, without relying on an intermediary server to provide special services. **Hawkins** does not disclose a single server that provides the recited steps. Instead, **Hawkins** discloses:

One embodiment of the invention includes a system having a distributed web site. The web site is distributed between a client, a server and a web server. The client stores a set of predefined applications that correspond to a part of the web site. The applications include data formatted according to a first markup language. From the set of predefined applications, the client can generate queries. The server receives the queries and generates new, related queries. The new queries correspond to a second query protocol. The second query protocol is used by the web server. The web server generates responses to the new queries and sends these responses to the server. The responses are formatted according to a second markup language. These responses correspond to the second portion of the web site. The server then converts the responses into new responses that the client can use.

Hawkins, column 3, lines 33-47

This excerpt discloses that the functions of a web site are distributed across two servers and the client. For instance, a query page where the user can request a specific piece of information would normally be stored on a web server and delivered to the user when the user enters the web site. **Hawkins** discloses that such a page is preferably included as “*pre-loaded content*” (column 12, line 32) on the client device. Even the rejection acknowledges that two different servers are involved in the process shown in **Hawkins**, noting that “*the server is shown at reference sign 140 (described as ‘Web Server’) ... storing a document at reference sign 144*” and “*Hawkins recites: ‘server 180 carries most of the burden of bringing the information’*”. Thus, **Hawkins** does not disclose a single server that will perform the claimed steps of “*receiving a request ... parsing the original JSP file for JSP tags; masking the JSP tags;*

converting non-masked tags ... unmasking the JSP tags; and storing a transformed JSP file". Therefore **Hawkins** does not meet this feature of the claims. Additionally, **Hunter** and **Ginter**, when considered as a whole, do not disclose or suggest a single server performing the steps of claim 1, nor has the Examiner shown otherwise. Therefore, since none of the cited references disclose this feature, a proper *prima facie* case of obviousness has not been made against claim 1.

File to be Converted Does Not Contain Executable Program Code

The rejection asserts that "*Hawkins discloses that the file contains executable program code*", apparently referring to "the file" that is being transformed in the claim. However, the reference to applets in **Hawkins** is in the context of very specialized programs that have little or nothing in common with the claimed JSPs. A more complete quote from **Hawkins** states:

Alternatively, some programs are customized for accessing specific information from particular web sites. Examples of these programs are Java applets that reside on the client or are served to the client by a server. The applets can then be reused to access information from a web site. An example of a specialized program for accessing specific information is the RealVideo Player from Real Networks, Inc. A problem with these types of programs is that they are very specific to a particular type of content. For example, they do not use standard HTML (hypertext markup language) constructs. This means that web site developers cannot use standard web site development tools to create their sites.

Hawkins, column 3, lines 14-25

Hawkins refers to exemplary applets "*that reside on the client or are served to the client*", so that the disclosed applets are executed on the client. This teaching is in contrast to both a servlet and the claimed JSPs, which are executed on the server. Thus, in the context of delivering a web page to a client, as the claimed invention is doing, it is incorrect to equate an applet executing on the client and a servlet executing on the server. Even if the disclosed applet were incorporated into the method of **Hawkins**, an applet on the client would not be involved in the transformation of a web page that is recited in the claims. Further, **Hawkins** does not disclose or suggest that the mentioned applets be incorporated into the methodology that this reference discloses. Additionally, although **Hunter** and **Ginter** are not cited to disclose this feature, neither **Hunter** nor **Ginter**, when considered as a whole, disclose or suggest that the file being transformed contains executable program code, nor has the rejection asserted otherwise. Since none of the cited references disclose or suggest this feature, a proper *prima facie* case of obviousness has not been made against claim 1.

Unmasking of JSP Tags Is Not Shown

The rejection asserts that "*Ginter teaches the use of masking tags*". However, neither **Ginter** nor the proposed combination considered as a whole discloses or suggests the step of "*unmasking the JSP tags*". The excerpt of **Ginter** cited against the use of masking states:

UDEs 1200 are preferably encrypted using a site specific key once they are loaded into a site. This site-specific key masks a validation tag that may be derived from a cryptographically strong pseudo-random sequence by the SPE 503 and updated each time the record is written back to the 40 secure database 610. This technique provides reasonable assurance that the UDE 1200 has not been tampered with nor substituted when it is requested by the system for the next use.

Ginter, column 150, lines 34-42

A JSP tag must not be masked if the tag is to be used to retrieve dynamic information for the JSP page. Although the above excerpt mentions that a validation tag is masked by the site-specific key, **Ginter** does not disclose or suggest that a tag is later unmasked, nor does the rejection provide any assertion that this step is shown by **Ginter**. Additionally, the rejection admits that this feature is not disclosed or suggested by **Hawkins** or **Hunter**. Therefore, since none of the references relied on disclose or suggest the feature of “*unmasking the JSP tags*”, the combination of the references when considered as a whole does not teach or suggest all of the features of the claims. Accordingly, no *prima facie* case of obviousness has been shown against claim 1.

References Do Not Teach Combination

In addition to the claimed features that are not shown by the references cited, no proper motivation has been given for the combination, as Applicants will show. The only conversion step discussed in **Hawkins** is the conversion to binary prior to sending the information to the client. Such a conversion to a completely different format is not conducive to masking of certain elements of the file, since masking to avoid conversion would prevent the extreme compression that **Hawkins** desires.

Although **Hawkins** discloses sending PvC device-dependent pages to a PvC device, this reference does not disclose the conversion of the pages from a general HTML format to a PvC-dependent format. Instead, this reference discloses the use of “*specially constructed, HTML pages. ... [For], example [Figure 1], the wireless communications device 100 is accessing the special HTML pages (e.g., HTML page 144)*” (**Hawkins**, column 10, lines 56-67, emphasis added). The conversion shown in **Hawkins** is the conversion to a highly compressed binary form just prior to sending the information to the device. **Hawkins** teaches that all tags are converted to binary in order to compress them, noting:

The wireless client 405 and the proxy server 180 use a special format for transferring screen 101 contents from the proxy server 180 to the wireless client 405 ... named Compact Markup Language (CML) ... CML is a stream of text and image data with imbedded formatting commands (tags). The tags are imbedded as binary data and hence are very compact.

Hawkins, column 21, line 33 through column 22, line 22

As Applicants have proved, the idea of masking some of the tags to protect them from the conversion to binary would go directly against the high compression of transmitted information that **Hawkins** teaches. The Examiner has responded to this argument with the following:

In response the examiner would point out that applicants claimed invention fails to disclose a file-transferring step, and second that the Hawkins' invention is directed toward conversion of files for use on pervasive computing devices within a low-bandwidth environment (where the low bandwidth issue is overcome by the compression step).

Office Action mailed June 7, 2006, page 11.

Applicants note that the presence or absence of a file-transferring step in the claims is irrelevant. Before a web page composed of a JSP is sent to the user, the servlet portions of the page are replaced with the dynamic data that they are designed to retrieve. Only in the server will the JSPs exist. Additionally, even if **Hawkins** converts files for use on a pervasive computing device, this fact is not enough to read on the invention as claimed; **Hawkins** must perform the same type of conversion in the same manner. Since the only conversion that **Hawkins** discusses is the conversion to binary, this reference does not disclose the claimed conversion, even in combination with the other references.

Applicants have demonstrated that several elements of claim 1 have not been shown in the references cited. Specifically, neither **Hawkins** nor **Hunter** disclose or suggest the use of JSPs in the conversion process of **Hawkins** and **Hawkins** does not teach any details regarding the conversion of regular web pages into web pages for small devices. Applicants have also demonstrated the lack of a proper motivation to combine at least **Hawkins' conversion** with **Ginter's masking**. The rejection of claim 1 is overcome.

Claims 8, 15, and 22 have been rejected for reasons similar to claim 1; their rejections are also overcome. The remaining claims are dependent on one of claims 1, 8, 15, and 22, so their rejections are also overcome. Therefore, the rejection of claims 1, 6, 7, 8, 13, 14, 15, 20-22, 27 and 28 under 35 U.S.C. § 103 has been overcome.

II. 35 U.S.C. § 103, Obviousness: Claims 2, 3, 9, 10, 16, 17, 23 and 24

Claims 2, 3, 9, 10, 16, 17, 23 and 24 stand rejected under 35 U.S.C. § 103 as being unpatentable over **Hawkins** in view of **Hunter**, **Ginter**, and **Judson**, Content Display During Idle Time as a User Waits for Information During an Internet Transaction, U.S. Patent No. 6,185,586 (February 6, 2001) (hereinafter "**Judson**"). This rejection is respectfully traversed.

Claims 9-10, 16-17 and 23-24 stand rejected under 35 U.S.C. § 103 as being unpatentable over **Hawkins** in view of **Hunter**, **Ginter**, and **Ramaley et al.**, Method for Managing Embedded Files for a

Document Saved in HTML Format, U.S. Patent No. 6,585,777 (July 1, 2003) (hereinafter “**Ramaley**”).

This rejection is respectfully traversed.

Claims 29-31 stand rejected under 35 U.S.C. § 103 as being unpatentable over **Hawkins** in view of **Hunter, Ginter, and Toyouchi** et al., Service Providing System and Method Which Divides a Request into Plural Service Requests and Provides an Integrated Service Based on Service Utilization History Information in Response to the Request, U.S. Patent No. 6,847,988 (January 25, 2005) (hereinafter “**Toyouchi**”). This rejection is respectfully traversed.

Judson, Ramaley, nor Toyouchi do not make up for the deficiencies of **Hawkins** and **Hunter**. All of the claims in these three rejections are dependent on one of the claims in the earlier rejection. Since the rejection of the independent claims has been shown to be overcome, the rejection of these claims is also overcome. Specifically, **Hawkins** does not disclose or suggest a single server performing the recited steps. **Hawkins** also does not disclose or suggest that the file to be converted contains JSP pages or their equivalent. Further, the references do not supply a proper motivation for their combination in order to meet the claimed invention. Additionally, neither **Judson, Ramaley, nor Toyouchi** make up for the deficiencies of **Hawkins** and **Hunter**.

Additionally, one of ordinary skill would not combine the references to achieve the invention of claim 1 because the references are directed towards solving different problems. It is necessary to consider the reality of the circumstances--in other words, common sense--in deciding in which fields a person of ordinary skill would reasonably be expected to look for a solution to the problem facing the inventor. *In re Oetiker*, 977 F.2d 1443 (Fed. Cir. 1992); *In re Wood*, 599 F.2d 1032, 1036, 202 U.S.P.Q. 171, 174 (CCPA 1979). In the case at hand, the cited references address distinct problems. Thus, no common sense reason exists to establish that one of ordinary skill would reasonably be expected to look for a solution to the problem facing the inventor. Accordingly, no teaching, suggestion, or motivation exists to combine the references and the examiner has failed to state a *prima facie* obviousness rejection of claim 1.

For example, **Hawkins** is directed to solving the problem of transmitting information on the Internet to a device that can only receive a relatively low bandwidth. For example, **Hawkins** provides that:

One area in which Web access is becoming more desirable is in handheld devices. Handheld devices are emerging as important computer devices. Handheld devices typically implement a relatively small, but important function set. Examples of such handheld devices are the PalmPilot.TM. handheld device available from 3COM Corporation, Inc. of Santa Clara, Calif. Examples of the function set supported are address books, calendars, and task lists. ...

An issue with using handheld devices to access the Web is related to their capabilities. Even if connected to a high bandwidth network, most handheld devices do not have the screen area or the processing power to display the graphics and large amounts of text in a typical web page. However, it is still desirable to support the browsing of information on the Web using handheld devices. It is further desirable that the handheld devices be able to use networks

that have relatively low bandwidths. ...

Therefore what is desired is an improved system and method for handheld device to access Internet information over relative low bandwidth networks.

Hawkins, column 2, line 35 through column 3, line 28.

On the other hand, **Ginter** is directed to the problem of protecting electronic content. For example **Ginter** provides as follows:

Telecommunications, financial transactions, government processes, business operations, entertainment, and personal business productivity all now depend on electronic appliances. Millions of these electronic appliances have been electronically connected together. These interconnected electronic appliances comprise what is increasingly called the "information highway." Many businesses, academicians, and government leaders are concerned about how to protect the rights of citizens and organizations who use this information (also "electronic" or "digital") highway.

Ginter, column 1, lines 41-50.

In still further contrast, **Judson** is directed to the problem of providing a user with information or activities that utilize the "downtime" that occurs between linking to a hypertext document and the final display of the document. For example, **Judson** provides as follows:

There is a finite time period between the time the user initiates the link and the return of the web page. Even when the web page is returned quickly, there is an additional time period during which formatting information must be processed for display on the display interface. For example, most web browsers display inline images (namely images next to text) using an X bit map (XBM) or .gif format. Each image takes time to process and slows down the initial display of the document. The user typically "sees" an essentially unrecognizable "image" on the display screen which only gradually comes into focus. It is only after the entire image is downloaded from the server and then processed by the browser that the user can fully access the web page itself. This "waiting" period is even longer when the client machine has a relatively slow modem, and often the user will have to wait many seconds before being able to see the in-line image and/or begin using the web page. This problem will be exacerbated when the next generation browser technology (such as Netscape Navigator 2.0) becomes more widely implemented because such browsers are being designed to handle much more complex download formats (for more interactive, dynamic displays).

Judson, column 1, lines 40-61.

In still further contrast, **Ramaley** is directed to the problem of managing embedded content in HTML format. For example, **Ramaley** provides as follows:

Users have clear expectations of how embedded content management should work. These expectations have been established from years of using traditional desktop productivity tools, such as word processing programs, which typically enable both embedding content in a primary document and editing the embedded content. In contrast, for Hyper Text Mark-up Language (HTML)-formatted documents, such as web pages, each piece of content is required to be a separate linked file. In other words, HTML does not directly support the concept of embedding content in the primary document. Nevertheless, the expectations of users have not changed in this HTML-formatted document environment because

they still desire HTML documents to support the characteristics of embedded content.

Ramaley, column 1, lines 15-28.

In still further contrast, **Toyouchi** is directed to the problem of providing end users with the capability to effectively utilize information services. For example, **Toyouchi** provides as follows:

(1) It is difficult to find out a service desired by an end user. In other words, even when a certain information providing computer provides a service desired by the end user, if the end user does not know of the existence of this information providing computer and/or connection destinations, then the end user cannot utilize this service. It is practically difficult for the end user to personally grasp a huge number of service contents of information providing computers and a large number of connection destinations. Thus, the end user can hardly utilize his truly desirable services.

(2) It is difficult to utilize a suitable service, depending upon a change in conditions. In other words, not only a large number of information providers enter into information service businesses every day, but also the presently available services of the existing information providing computers are frequently changed and/or added with other services every day. End users can hardly, personally grasp all of these changes. As a result, even though it is highly likely that better services are newly added, the end users may fixedly utilizing specific services. Also system environments and qualities of information/services may change contents of services which can give higher satisfaction to the requests of the end users. It is practically difficult that the end users systematically evaluate these conditions and thus select the proper services in accordance with this change.

(3) It is difficult to discriminate services from each other based on desires and service utilization histories of end users. The desires and service utilization histories of end users are grasped only by these end users themselves. As a consequence, the information providing computers cannot select the proper services which can provide the higher rate of satisfaction to the requests of the respective end users, but therefore the end users themselves must retrieve their desirable services and also must hold/manage the service utilization histories.

(4) When services of a plurality of information providing computers are integrated to be received, procedures by end users become cumbersome. That is, when a complex service provided by a plurality of information providing computers is received, an end user must be sequentially connected with the respective information providing computers, must retrieve the contents of the individual services, must propose the individual service utilizations, and must adjust these services individually. Also, the end user is required to pay his fee to the received services from the respective information providers. In particular, an abundance of time and workloads on the part of the end user are required so as to adjust combinations of the plural services.

Toyouchi, column 1, line 60 through column 2, line 41.

Based on the plain disclosures of the references themselves, the references address completely distinct problems that are unrelated to each other. The problem of transmitting information to a low-bandwidth device is completely distinct from the problem of protecting electronic content. Similarly, both of these problems are completely distinct from the problem of providing a user with downtime activities, as

disclosed in **Judson**. Likewise, these problems are completely distinct from the problem of managing embedded content in HTML format, as disclosed in **Ramaley**. Likewise, these problems are completely distinct from the problem of helping end users to effectively utilize information services, as disclosed in **Toyouchi**.

Because the references address completely distinct problems, one of ordinary skill would have no reason to combine or otherwise modify the references to achieve the inventions recited in these claims. Thus, no proper teaching, suggestion, or motivation exists to combine the references in the manner suggested by the examiner. Accordingly, the examiner has failed to state a *prima facie* obviousness rejection against claim 1 or any other claim in this grouping of claims.

Therefore, each of these rejections is overcome.

III. Conclusion

It is respectfully urged that the subject application is patentable over the cited references and is now in condition for allowance. The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: September 7, 2006

Respectfully submitted,

/Betty Formby/

Betty Formby
Reg. No. 36,536
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Agent for Applicants